Artificiell Intelligens för Civilingenjörer

(DT510G)

Sample Exam with Solutions

Teacher:	Alessandro Saffiotti
Date:	2021-12-06
No. exercises:	5
Total points:	110 (61 required to pass)

This is an example of a solved assignment for the first exam of the HT2020 edition of this course. This example is meant to give you a hint of the type of questions and exercises that you may expect at your exam. It also contains examples of answers to the questions and exercises, to help you test your knowledge. Needless to say, the actual questions and exercises that you will be given, as well as their number, their topics and the maximum points they give you, may be different.

Exercise 1 (30 points)

Answer the following statements with either True or False. No answer means 0 points, a wrong answer counts for -1 point. A correct answer gives you $+1$ point, but if you add a correct and detailed explanation you get 2 or 3 points, depending on the quality of the explanation. If your total in this exercise is negative, the whole exercise will be counted as 0 points.

S1: Suppose that you are given an ontology in Description Logic that includes the following clauses:

$$\begin{array}{l}\text{Coronavirus} \sqsubseteq \text{Virus}\\ \text{Ebolavirus} \sqsubseteq \text{Virus}\\ \text{Deadly-Virus} \equiv \text{Virus} \sqcap \exists\text{hasEffect.DeadlyDisease}\\ \text{EBOV : Ebolavirus}\\ \text{EBOV hasEffect KidneyInjury}\end{array}$$

From this ontology one can infer that EBOV is an instance of a Deadly-Virus. (Why?).

False: we also need the fact "KidneyInjury : DeadlyDisease" to infer it.

S2: In fuzzy set theory, when we fuse information from different sources we do not find an "average" but rather an "agreement" among the sources. (Why?)

True: we use fuzzy set intersection to find options that are possible according to all sources of information, and the degree of this possibility.

S3: Data-driven approaches to AI typically rely on a model extracted from knowledge provided by human experts. (Why?)

False: this is done in model-based approaches to AI. Data-driven approaches typically use a large amount of data to automatically generate a model.

S4: In constraint satisfaction problems, *least constraining value* (LCV) is a heuristic to select the next variable. (Why?).

False: it is a value selection heuristic, that is, a heuristic to select the next value to try for a given variable.

S5: In a neural network, forward propagation and backward propagation are both used during training. (Why?)

True: forward propagation computes the output using the current weights; backward propagation updates the weights, considering the error between the computed output and the target output.

S6: K-means is a classification technique. (Why?)

False: K-means is a clustering technique, which is a type of unsupervised learning. Classification, by contrast, is a type of supervised learning (although some authors talk of "unsupervised classification").

S7: In a neural network, we use forward propagation if we want faster computation, and we use backward propagation if we want more accurate results. (Why?)

False: forward propagation is used to propagate values from the input to the output; backward propagation is used to propagate the error value from the output and update the weights in the network. Both are needed during training to improve accuracy, and are not related to speed.

S8: Probability theory can be easily used to add the ability to deal with uncertainty to a rule-based system. (Why?)

False: probability theory is not truth-functional, which makes it difficult to use in logic-based and rule-based systems, that are truth-functional.

S9: Data-driven approaches to AI typically extract a model from a large set of data. (Why?)

True: this contrasts with model-based approaches to AI, which typically rely on a model extracted from knowledge provided by human experts.

S10: Fuzzy set theory can be seen as a special case of probability theory. (Why?)

False: fuzzy set theory is meant to represent a different type of imprecise knowledge, namely, vague knowledge as opposed to statistical knowledge.

Exercise 2 (15 points)

The A* search algorithm expands, at each step, the node in the *frontier* that has the lowest F-VALUE. Consider an algorithm, called *A, which is identical to A* except that it expands, at each step, the node in the *frontier* that has the highest F-VALUE.
Answer shortly but clearly each of the following questions:

a) Is the *A algorithm complete? Explain why.

b) Is the *A algorithm optimal? Explain why.

c) What are the time and the space complexities of the *A algorithm? Explain your answer.

Answers:

a) (max 5 points) Like all heuristic search strategies, it is complete if the search space is finite, since eventually all the space will be explored and a solution will be found if there is one. Completeness is not guaranteed, though, if the search space is not finite: the algorithm may stay forever in a part of the space different from the one that contains the solution.

b) (max 5 points) It is not optimal. Recall that the proof of optimality of A* exploits the fact that A* will never expand a goal node that has a higher F-VALUE than any node on an optimal path. By chosing the node in the frontier with the highest F-VALUE, the given *A algorithm violates this condition. You can build a minimal counter-example using the graph from Lecture 2, part 4, slide 13, and letting $h(n) = 0$ for every node.

c) (max 5 points) Time and space complexity are the same as for heuristic greedy search. Time complexity is $O(b^m)$ since in the worse case we need to expand all the nodes in the search tree before we find a goal one. Space complexity is $O(bm)$ since in the worse case the frontier will contain all the nodes in the curent path, which are at most m, and all their unexplored children, which are at most b for each node.

Exercise 3 (25 points)

Consider the following problem.

> *In Penny Lane there are three houses in a row. Each one contains a person of a different nationality, who drinks a different drink, and owns a different pet. The following facts are known:*
>
> 1. *The person in the third house drinks milk*
> 2. *The Spaniard drinks coffee*
> 3. *The Norwegian owns the parrot*
> 4. *The Ukranian lives next to the coffee drinker*
> 5. *The tea drinker owns the rabbit.*
>
> *Who owns the hamster?*

Answer shortly but clearly each of the following questions:

a) Cast the above problem as a CSP. In particular, you need to write down: what are the variables $\{X_1, X_2, \ldots, X_n\}$, what are the respective domains $\{D_1, D_2, \ldots, D_n\}$, and what are the constraints $\{C_1, C_2, \ldots, C_k\}$.

b) Suppose you want to solve the CSP from question (a) using a backtracking algorithtm. What variable would be selected first, according to the MRV heuristic? Explain your reasoning.

c) Continued from the previous question: show all the steps done by a backtracking algorithm to solve the CSP from question (a), until a solution is found.

d) In the above question, show at which step the algorithm could use forward checking to terminate a branch: explain your reasoning.

Answers:

a) (max 5 points) We can use one variable for each of the nine features, indicating in which house the feature is hosted. Namely: one variables for each national (Norwegian, Spaniard, Ukranian), one variable for each drink (milk, coffee, tea), one variable for each pet (parrot, rabbit, hamster). Domain for each variable: $\{1, 2, 3\}$ (house number). Constraints: milk $= 3$, Spaniard $=$ coffee, Norwegian $=$ parrot, tea $=$ rabbit, Ukranian $=$ coffee± 1. Plus the constraints that all pets, all nationals and all drinks are in different houses, that is, they have different values: Norwegian $\neq$ Spaniard, Spaniard $\neq$ Ukranian, Norwegian $\neq$ Ukranian; milk $\neq$ coffee, and so on.
(Remark: there may be other ways to cast this problem as a CSP, the important things are to follow the definition of CSP and be precise.)

b) (max 5 points) Trivially, the most constrainted variable is milk = 3, and this variable is assigned at start. Then, both variables coffee and tea only have two possible values (1 or 2) while all others have three: therefore, either one of these two variables would be selected next according to the MRV heuristic.

c) (max 10 points) Suppose variable coffee is selected. We first try to assign value 1 to it. Now three variables have only one possible value: tea (2), Spaniard (1), rabbit (2). Using MRV, we select these three variables one after the other and assign them their only possible value. Then, MRV selects Norwegian that has two possible values: 2 or 3. We first try to assign value 2. Now, the variable parrot must have the same value, that is 2, but this value has already been assigned to rabbit, violating the constraint that all pets live in different houses. We therefore backtrack on the assignment to Norwegian, and try to assign value 3 to it. Now parrot must have the same value 3, and the variable hamster can only have one value: 1. So, the hamster and the Spaniard are in the same house, or said differently, the Spaniard owns the hamster, which is the answer to our problem.

d) (max 5 points) When trying to assign 2 to Norwegian, a step of forward checking constraint propagation would show that the domain of the variable parrot is empty. Therefore, we can skip that value and prune the corresponding branch of the search tree, assigning instead 3 to Norwegian.

Exercise 4 (25 points)

Assume that you have the following STRIPS-style planning operator, to put an animal `?x` in a cage `?y`, provided that both `?x` and `?y` are at the same place `?z`:

```
(:action put-in-cage
   :param (?x ?y ?z)
   :precond (and (at ?x ?z) (at ?y ?z))
   :effect (and (inside ?x ?y) (not (at ?x ?z)))
)
```

Then, assume that you have a second operator, to take `?x` out of the cage `?y`:

```
(:action take-from-cage
   :param (?x ?y ?z)
   :precond (and (inside ?x ?y) (at ?y ?z))
   :effect (and (at ?x ?z) (not (inside ?x ?y)))
)
```

Finally, assume this initial state: `((at cage house2) (at hamster house2))`
and this goal state: `(at hamster house1)`
Answer shortly but clearly each of the following questions:

a) Tell whether or not the operator `put-in-cage` is applicable in the initial state, with the `?x` parameter bound to hamster, `?y` bound to cage, and `?z` bound to house2. If yes, write the resulting state after it is applied. If not, explain why.

b) Tell whether or not the operator `take-from-cage` is applicable in the initial state, with all the parameters bound as above. If yes, write the resulting state after it is applied. If not, explain why.

c) Tell whether or not you can use the above operators to generate a plan to go from the given initial state to the given goal state, that is, to move the hamster from house2 to house1. If yes, explain how. If not, explain why.

d) If the answer to question (c) is "no", write one additional STRIPS-style planning operator so that you can solve the above planning problem. Explain the operator, and show the resulting plan.

Answers:

a) (max 5 points) Yes, it is applicable. In fact, with the given bindings, the precondition of the operator becomes: `(and (at hamster house2) (at cage house2))`. Both literals in the conjunction are in the initial state, therefore the precondition holds and the operator can be applied.
After the operator is applied, the state becomes: `((inside hamster cage)`

`(at cage house2))`. The literal `(inside hamster cage)` has been added to the state because it appears in the `:effect` field of the operator. The literal `(at hamster house2)` has been removed from the state because it appears negated in the `:effect` field. The other literal in the initial state, `(at cage house2)`, does not appear in the `:effect` field; therefore, by virtue of the frame axiom, it is maintained after the operator is applied.

b) (max 5 points) No, it is not applicable. With these bindings, the precondition of the operator is: `(and (inside hamster cage) (at cage house2))`. Since the literal `(inside hamster cage)` is not in the initial state, the precondition does not hold and the operator cannot be applied.

c) (max 5 points) No, we can not. To make the literal `(at hamster house1)` true, as requested in the goal state, we need to apply `take-from-cage` operator with the parameter `?z` bound to house1. To do so, we must be in a state where the precondition `(at cage house1)` is true. But in the initial state, we have `(at cage house2)`, and we do not have any operator that can change this into a state where `(at cage house1)` is true.

d) (max 10 points) Following up from the previous answer, we need an operator to move the cage from one place to another one. The following one would do the job:

```
( :action move-cage
   :param (?x ?y ?z)
   :precond (at ?x ?y)
   :effect (and (at ?x ?z) (not (at ?x ?y))) )
```

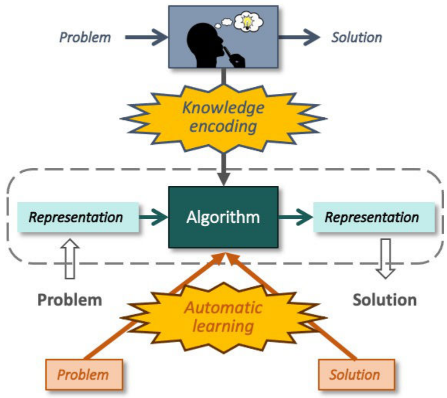
If we apply the operator `put-in-cage` from the initial state, we obtain the state `((inside hamster cage) (at cage house2))`. If we apply the new operator `move-cage` from this state, with the parameter `?z` bound to `house1`, we obtain the state `((inside hamster cage) (at cage house1))`. Finally, if we apply the operator `take-from-cage` from this last state, we obtain the state `((at hamster house1) (at cage house1))`, and the goal is true in this state. The overall plan is therefore:

```
( (put-in-cage hamster cage house2)
  (move-cage cage house2 house1)
  (take-from-cage hamster cage house1) )
```

Remark: the above operator `move-cage` can in principle be used to move directly the hamster, by binding `?x` to hamster. This, however, should be avoided, since it would fail to capture the fact that we must use a cage to transport pets. One way to avoid this is to limit the applicability of the operator, e.g., by adding a precondition like: `(is-cage ?x)`.

Exercise 5 (15 points)

In our course, we have made a distinction between "Knowledge-based" and "Data-driven" approaches to Artificial Intelligence. We have illustrated the difference between these two classes of approaches using the figure on the right.



Answer shortly but clearly each of the following questions:

a) Explain, in your own words, the difference between knowlege-based approaches and data-driven approaches to AI. You may refer to the above figure in your explanation, if you want.

b) Explain, in your own words, the main limitations of knowlege-based approaches to AI. Give an example of a (real or imagined) application where it would be difficult to use knowlege-based approaches because of these limitations, and explain why.

c) Explain, in your own words, the main limitations of data-driven approaches to AI. Give an example of a (real or imagined) application where it would be difficult to use data-driven approaches because of these limitations, and explain why.

Answers: This is an open question, everyone may phrase the answer differently. All the information to answer this question was given in Lecture 8, part 3.